MINEX
An Evaluation-based Program for the Improvement of Minutiae Interoperability

MINEX II

Interoperability of the ISO/IEC 19794-2 Compact Card and ISO/IEC 7816-11 Match-on-Card Specifications

Patrick Grother and Wayne Salamon
NIST

May 25, 2007

Version History
May 24, 2007: This interim version is available for comment until May 31, 2007. The changes from May 8 are shown in blue.
May 8, 2007. NIST is pleased to announce the availability of this draft evaluation plan. The entire content of this document is subject to revision. NIST specifically requests reviewers of the document to submit comments to the authors. These may include suggestions to add, subtract, or change its content.
March 12, 2007. NIST requested comments on the feasibility of MOC. NIST is most appreciative of the numerous, enthusiastic and helpful responses.

#	Question to reviewers
1.	Should NIST require all matcher providers to specify an operating threshold at the time they submit the card?
2.	Regarding Table 12, is there any utility in the '80' and '83' tags in each BIT? ISO/IEC 7816-11 Table C.1 is somewhat terse on this matter!
3.	How should NIST recover card version information?
4.	How should NIST recover matching algorithm version information?
5.	Regarding the four byte view header in Table 4 and Table 5, NIST solicits opinion on whether any of the fields are useful. Particularly – How would finger position be transmitted in an operational two or more finger application? – Is the number of minutia inferred from the length-of-data field?
6.	The document does not currently allow for storage of a proprietary or non-standard reference template on the card. Should this be included, or prioritized?

1
2
3

Candidate Timeline

Sep 17-21	Phase 2 submission period	Tentative
Sep 11-13	Biometrics Consortium, Baltimore MD	
Early Sep	Workshop to discuss Phase 1	
Aug 17	Complete return of Phase 1 results to suppliers	
Jul 16-Aug 3	Phase 1 submission period	
June 29	Definitive Test Plan (after SC 37 Berlin)	
June 15	Release of draft test plan, opportunity for comments	
May 31	Comment period closes	
May 24	sBMOC Workshop at NIST	
May 9	First draft MINEX II, ISO-CC + MOC Test Plan	
March 29	Comment period closed	
March 12	MOC Concept document + RFP	

Table of Contents

1. Overview of the MINEX tests 6
2. Scope 6
3. Caution 7
3.1. Relationship between MINEX II and other issues 7
3.2. Relationship between MINEX II and Ongoing MINEX 8
4. Abbreviations 8
5. Aspects of the test 8
5.1. Match-on-card vs. match-off-card 8
5.2. Fundamental concept of the test 9
5.3. Card-matcher combinations 9
5.4. Generic interfaces 9
5.5. Three-way interoperability 9
5.6. Two phase testing 11
5.7. Options for participation 11
5.8. Specification of an operating point 11
6. Profile of ISO/IEC 19794-2 compact card 12
6.1. Record structure 12
6.2. Use of header 13
6.3. Record length 13
6.4. Failure to acquire 13
7. MOC interface specification 14
7.1. Selection of the test application 14
7.2. Store enrollment template on the card 14
7.3. NIST read of the BIT 15
7.4. NIST use of the BIT 17
7.5. Verification 19
7.6. Reading card identifier 20
7.7. Reading matcher identifier 21
8. Comparison of the INCITS 378 and ISO/IEC 19794-2 standards 21
8.1. Syntactic differences 21
8.2. Semantic differences 22
8.3. Simulation protocol 22
8.4. Results after re-quantization of angle and position 23
8.5. Results after reduction in number of minutiae 24
8.6. Conclusions 26
9. Privacy requirements 27
9.1. Returning software to vendors 27
9.2. Returning cards to vendors 27
10. Ambiguities in the standards 27
11. API specification 27
11.1. Overview 27
11.2. Fingerprint image data 28
11.3. INCITS 800-76 compliant templates 29
11.4. Testing interface description 29
11.5. Software and Documentation 32
11.6. References 33
12. Application Form 33
12.1. Who Should Participate 33
12.2. How to Participate 33
12.3. NIST Activity 34
12.4. Points of Contact 34
12.5. Access to MINEX II Validation Data 34
12.6. Access to MINEX II Test Data 35
12.7. Reporting of Results 35

12.8. Agreement to Participate 35

List of Figures

Figure 1 – Native vs. Two-way vs. Three-way Interoperability 10
Figure 2 - Use of INCITS 378 for simulation of ISO-CC accuracy 22
Figure 3 – Effect of minutia pruning on FNMR (above) and FMR (below) 26
Figure 4 – Fingerprint image raster scan order 28

List of Tables

Table 1 – Family of MINEX evaluations 6
Table 2 – Abbreviations 8
Table 3 – MINEX II classes of participation 11
Table 4 – NIST profile of ISO/IEC 19794-2 profile 12
Table 5 – ISO/IEC 19794-2 minutiae template DO 13
Table 6 – Command APDU for selection of MOC application 14
Table 7 – Response APDU from selection of MOC application 14
Table 8 – Command APDU for storage of reference template 14
Table 9 – Response APDU from storage of reference template 15
Table 10 – Command APDU for retrieval of biometric information template 15
Table 11 – Response APDU from retrieval of biometric information template 15
Table 12 - ISO/IEC 19794-2 Biometric Information Template 15
Table 13 – Command APDU for comparison of biometric templates 19
Table 14 – Response APDU from comparison of biometric templates 19
Table 15 – Command APDU for retrieval of verification similarity score 19
Table 16 – Response APDU for retrieval of verification similarity score 20
Table 17 – Command APDU for retrieval of Card identifier 20
Table 18 – Response APDU for retrieval of Card identifier 21
Table 17 – Command APDU for retrieval of Matcher identifier 21
Table 18 – Response APDU for retrieval of Matcher identifier 21
Table 17 – Minutia encodings of the ISO-CC and INCITS 378 22
Table 18 – Minutia location quantization of ISO-CC and INCITS 378 templates 22
Table 19 – Minutia angle quantization of ISO-CC and INCITS 378 templates 22
Table 20 – Source code for conversion of INCITS 378 to ISO-CC 23
Table 21 – Relative accuracy of ISO-CC and INCITS 378 templates (Fixed FMR) 24
Table 22 – Relative accuracy of ISO-CC and INCITS 378 templates (fixed t) 24
Table 23 – MINEX predefined values 29
Table 24 – MINEX create_template API Function 29
Table 25 – MINEX API match_templates function 30
Table 26 – MINEX API get_pids function 31
Table 27 – MINEX API return codes 31

1. Overview of the MINEX tests

The MINEX program is intended to improve template-based interoperability from the state reported in MINEX 04[1] and MTIT[2] toward that achievable with image-based implementations. The approach is to conduct several trials, MINEX II, III, IV etc, each of which will embed development, evaluation, targeted feedback and consultation activities between NIST, industry and other interested parties. Within scope is anything to do with fingerprint minutiae as an interoperable biometric for identity management. Typical outcomes will be measurements of accuracy, processing time, template size, proposals to revise the relevant standards, studies of use of quality measures, calibration information, and new metrics.

Table 1 gives an overview of the various MINEX activities at NIST, and a context for future work, starting with MINEX II.

Table 1 – Family of MINEX evaluations

MINEX 04[1]	This test was conducted as an initial comparison of image vs. minutia-based interoperability. It assessed the core algorithmic ability of fingerprint matcher Z to compare minutiae templates from sources X and Y. It compared that case with fully proprietary templates on the same sets of archival data. The test adopted the INCITS 378 template as a base standard. The test is now closed. http://fingeprint.nist.gov/minex04
Ongoing MINEX[3]	Ongoing MINEX is a continuing program of interoperability assessment intended to measure conformance and interoperability of INCITS 378:2004 samples. The test uses one expanded partition of the MINEX 04 data to formulate interoperable groups of matchers and template generators. One client of Ongoing MINEX is the US Government's PIV program which has its own set of criteria against which the interoperable group is formed. The test results are available to other applications or programs which may elect to set their own criteria for interoperable performance. The test remains open.
MINEX II	MINEX II will consider the ISO/IEC 19794-2 compact card template, and Match-on-Card capabilities. The test design is in the formative stages: This document describes MINEX II and is open for comment. Readers should not consider any of the content as final.

2. Scope

MINEX II is intended to measure the core algorithmic capabilities of fingerprint matching algorithms running on standardized ISO/IEC 7816 smart cards. Specifically the test will

- instantiate a mechanism for MOC testing,
- measure the accuracy of match-on-card (MOC) and match-off-card (MXC) implementations,
- measure the accuracy of ISO/IEC 19794-2 compact card template generators and matchers,
- time the various operations,

[1] Minutiae Interoperability Exchange Test, MINEX, NIST Interagency Report 7296, March 21, 2006.
[2] Minutiae Template Interoperability Testing, http://www.mtitproject.com
[3] See the Ongoing MINEX Homepage at http://fingerprint.nist.gov/minex

- test the viability of INCITS 378 to ISO/IEC 19794-2 compact card transcoding, and
- formulate comments toward possible revision of the relevant standards.
- Conformance to the ISO/IEC 19794-2 compact card format, as profiled herein.

The primary outputs of the test will be statements of performance using the following metrics are:

- False non-match and false match error rates,
- Off-card template generation times,
- On-card matching times.

In so doing, this evaluation establishes

- a profile of the ISO/IEC 19794-2 standard, and
- an advised mechanism for transcoding INCITS 378 to ISO/IEC 19794-2 templates.

Not in the scope of this evaluation, but the subject of a separate activity at NIST is:

- securing the communications channel, including cryptographic protection of the biometric templates;
- protecting the integrity of the templates, including digital signatures;
- authentication of the card or the reader;
- timing of these operations;
- contactless communications;

The following are specifically not within the current scope of this evaluation:

- The ISO/IEC 19794-2 "record" and "card normal" templates;
- Evaluation of readers, including performance, conformance and interoperability;
- Evaluation of ruggedness or durability of the card;
- On-card template generation (i.e. extraction of minutiae from images);
- Template update or adaptation;
- Although the test will use ISO/IEC 7816 parts 4 and 11, and conformance to this subset will be a requirement, this study does not constitute a formal test of conformance to any part of ISO/IEC 7816;
- Devices not conforming to ISO/IEC 7816, including all sense-on-card devices that embed proprietary template formats.

DELETE: NIST is open to discussing the scope.

3. Caution

3.1. Relationship between MINEX II and other issues

Neither this document, nor any future execution of MINEX or MOC evaluations by NIST, should be construed as an indication that NIST, nor any other agency of the US government, has decided for or against the inclusion or exclusion of the items listed below in any current or future government specification or program.

- Contactless biometric interfaces,
- MOC implementations,
- ISO/IEC 19794-2 templates, and
- Record headers in standardized templates, stored or transmitted to cards.

This document is strictly a special notice that is being published to reach a wider audience. It is not a federal procurement action, and no RFQ or RFP is available.

In addition, the identification of any commercial product or trade name does not imply endorsement or recommendation by the National Institute of Standards and Technology.

3.2. Relationship between MINEX II and Ongoing MINEX

MINEX II will use the same corpus of nearly 750000 images as is used in the Ongoing MINEX process. The test will produce at least the same accuracy metrics of core algorithmic capability as is generated in Ongoing MINEX. The issue of equivalence between the two tests is therefore under consideration at NIST, particularly whether an implementation using ISO/IEC 19794-2 compact card templates will offer comparable performance on INCITS 378:2004 templates.

4. Abbreviations

The abbreviations and acronyms of Table 2 are used in many parts of this document

Table 2 – Abbreviations

APDU	Application Protocol Data Unit as used in ISO/IEC 7816-4
BIT	Biometric Information Template as defined in ISO/IEC 7816-11
IDMS	Identity management system
FMR	False match rate
FNMR	False non-match rate
ISO/IEC 7816	Multipart standard for "Identification cards - Integrated circuit(s) cards with contacts"
ISO/IEC 7816-4:2005	ISO/IEC 7816 part entitled "Interindustry commands for interchange"
ISO/IEC 7816-11:2004	ISO/IEC 7816 part entitled "Personal verification through biometric methods"
ISO/IEC 19794	Multipart standard of "Biometric data interchange formats"
ISO/IEC 19794-2:2005	ISO/IEC 19794 part entitled "Finger minutiae data"
ISO-CC	The compact card minutia format of clause 8 in ISO/IEC 19794-2
INCITS 378:2004	US Fingerprint minutia exchange standard, precursor to ISO/IEC 19794-2
MINEX	Generic name for the series of NIST's Minutia Interoperability Exchange Tests
MOC	Match on card
MXC	Match off card
NIST	National Institute of Standards and Technology
PC/SC	Generic interface specification for PC to smart card connectivity
SC 17	Subcommittee 37 of Joint Technical Committee 1 – developer of smart card standards
SC 37	Subcommittee 37 of Joint Technical Committee 1 – developer of biometric standards

5. Aspects of the test

5.1. Match-on-card vs. match-off-card

The test is not specifically a match-on-card (MOC) test. It is primarily an evaluation of the ISO/IEC 19794-2 compact card template (ISO-CC) as profiled in section 6. ISO-CC is under consideration for a number of applications.

The test is secondarily a test of matchers that may also have a card-based implementation. The ISO/IEC 19794-2 activity is part of the MINEX development process designed to improve interoperability between minutiae-based implementations. The MOC work is initiated in response to near-term imperatives to evaluate the technology.

The test is likely to require customized cards (i.e. not production cards), at least because of our requirement to be able to read similarity scores from the card.

5.2. Fundamental concept of the test

The MOC capability shall be tested as follows:

- Two ISO/IEC 7816-11 BITs will be read from the card as a group, and stored (see section 7.3).
- The matching algorithm on the card will be tested by running a cross-compiled or otherwise emulated version of it on standard Pentium-class machines. This phase will embed hundreds of thousands of genuine and impostor comparisons. All templates sent to the matcher will be processed according to the respective BIT (see section 7.4).
- Selected genuine and impostor comparisons will be repeated on the card by successively storing reference templates on the card, and sending identical the same pairs of verification templates to the card for comparison.
- The similarity scores from the PC-based phase will be compared with those from the card. They will be required to be identical. What is 1 in 10^n are not?

This three-phase strategy assures NIST that the accuracy of the MOC implementation is identical to that of the PC-based port of the algorithm. NIST will measure the elapsed times of these operations.

This test embeds techniques beyond those standardized in ISO/IEC FCD 19795-4 (Biometric Performance Testing and Reporting – Part 4: Interoperability Performance Testing) the provisions of which MINEX tests follow.

5.3. Card-matcher combinations

SHOULD NIST REQUIRE TEAMING or SUBCONTRACTING AGREEMENTS TO NOT BE DISCLOSED TO NIST OR CO-SIGNED SUBMISSIONS FROM TWO COMPANIES?

NIST intends to evaluate paired card-matching algorithm combinations. NIST anticipates that the same algorithm implemented on a more capable card will offer improved performance. NIST suppliers participating in the test may submit multiple entries, for example:

- A card vendor may elect to team with several fingerprint matcher vendors;
- A fingerprint vendor may elect to team with multiple card manufacturers.

NIST will require

- text versioning information for the card and software matcher,
- card and algorithm product identifiers to be readable by the mechanism described in forthcoming API, see section 7.6 and 7.7
- technical contact information in both organizations, and
- cross-signed applications from authorized responsible parties in both organizations.

5.4. Generic interfaces

NIST intends to access all cards via third party PC/SC hardware owned by NIST. NIST is likely to use the M.U.S.C.L.E open-source drivers[4] under Linux.

As in previous tests, NIST intends to run the PC-based portions of the test using software components implementing a simple "C" API.

REQUEST: The test description should specify the reader type/revision so that the card submitters can test that the card can negotiate relevant parameters (speed, etc.) correctly with the reader.

5.5. Three-way interoperability

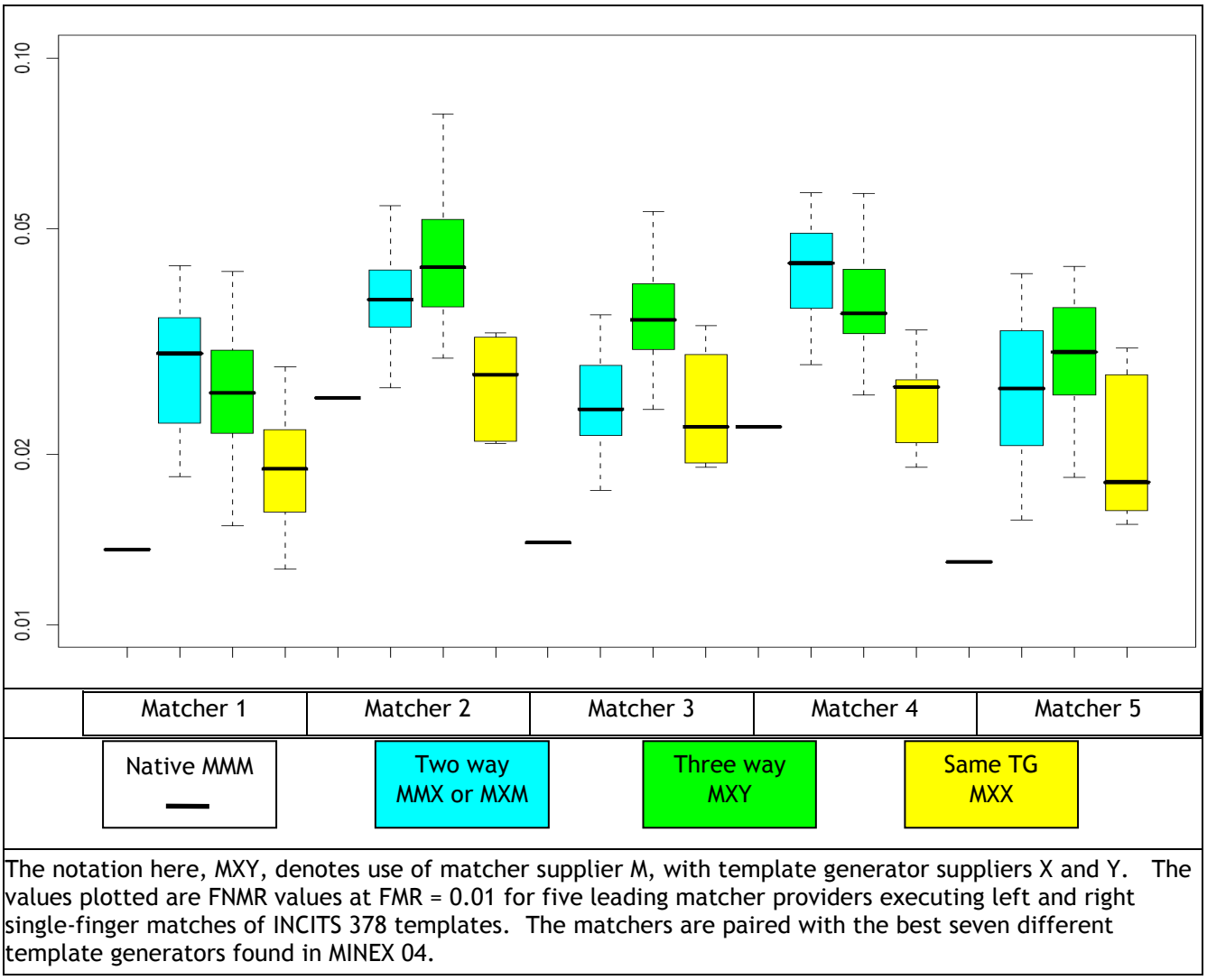
NIST anticipates that some cards embed fingerprint matchers that are not accompanied by an associated template generator, and may well be used with enrollment and verification templates coming from two

[4] See homepage http://www.linuxnet.com/middle.html

different sources. That is card-based matcher M, will compare a reference template produced by X with a verification template from Y. NIST will evaluate three-way interoperability in this test. MINEX participants should be aware that the MINEX 04 measurements of Figure 1, show three-way interoperability (green) will generally offer poorer performance

- than in two-way (cyan) or native (horizontal bar) cases, and
- in situations (matchers 1 and 4) in which the provider of a matcher supplies a template generator that is less effective than its peers.

Figure 1 – Native vs. Two-way vs. Three-way Interoperability



The notation here, MXY, denotes use of matcher supplier M, with template generator suppliers X and Y. The values plotted are FNMR values at FMR = 0.01 for five leading matcher providers executing left and right single-finger matches of INCITS 378 templates. The matchers are paired with the best seven different template generators found in MINEX 04.

NIST acknowledges that not requiring a card/matcher provider to team with a template generator provider is inconsistent with the requirement in section 5.3 to require pairing of card and matcher combinations. However this seeming inconsistency is undertaken in light of the following:

- Cards and matching algorithms may well be bound at time of manufacture;
- Cards and reference templates are bound at time the card issuance or IDMS software was shipped by an integrator;
- Although software matcher implementations (e.g. Java applets) exist (and are eligible for testing) and these could be selected during integration or even issuance, they are reported to run significantly more slowly.

Further NIST is motivated by a desire to improve performance, both in this evaluation and in fielded operation. Given that NIST has demonstrated that some template generators are better than others, NIST seeks to measure card-matcher-template generator compatibility.

5.6. Two phase testing

NIST is likely to conduct the test in two phases. The first is intended to be a small and fast evaluation of submitted software and/or cards that

- will use a reduced amount of data,
- will not consider all interoperability paradigms (see previous subsection),
- will give feedback and results to the suppliers,
- is not intended to include release of results to the sponsors or the public.

The second phase will be the full size test and will result in a final public report. The names of participants in both Phase 1 and Phase 2 will be reported in the Phase 2 report.

NIST is particularly interested in facilitating improved performance between Phase 1 and Phase 2. It intends to provide feedback to suppliers and to allow submission of improved software and hardware. NIST solicits comment on what feedback (metrics, data) would be most useful. NIST may conduct a workshop between Phases 1 and 2 to discuss this and other issues. WORKSHOP SEPTEMBER TBD

5.7. Options for participation

Participants must provide one of the combinations of components enumerated in Table 3.

Table 3 – MINEX II classes of participation

Class of Participation	Participation form agreeing to NIST conditions	ISO/IEC 7816 Card + Matcher	Software Template Matcher	Software ISO-CC Template Generator	
Class A	+	+	+		
Class B	+	+	+	+	
Class C	+		+	+	
Class D	+			+	

NIST does not intend to limit participation to one class only. So participants may submit entirely separate class A and class C submissions, for example.

DELETE: NIST will consider requests for addition or removal or some classes of participation.

5.8. Specification of an operating point

FOR DISCUSSION: Algorithms working on a FAR level of 1:100 usually need to be built differently from an algorithm working on a level of 1:10,000 (especially for smart cards where limitations in memory /computing power constrains what you can do) so it is not a fair comparison to plot the ROC curve and from it determine the FRR for different FAR values.

… it is important that the test specifies the operational threshold. We suggest a target of a FAR = 1:10,000, or create different classes, e.g. 1:10,000 and 1:100, and require the matcher provider to specify which classes they support.

NIST will report ROCs in any case. But in addition three options:

1	Ignore the request.
2	Require provision of matchers specifically for FMR = 0.0001 and 0.001 and 0.01 On separate cards?
3	Some NIST evaluations have defined a cost function as the definitive reporting variable: Cost = Cimp * Pimp * FMR + Cgen * Pgen * FNMR With explicit pre-test specification of, specific values, for example: Priors: Pimp = 0.001, Pgen = 0.999 Costs: Cimp = 10, Cgen = 1

6. Profile of ISO/IEC 19794-2 compact card

6.1. Record structure

This section defines precisely what constitutes an ISO-CC template in the NIST evaluation. It is included here because ISO/IEC 19794-2 is not clear on whether the card minutiae data should follow a header. Working Group 3 of SC 37 discussed this issue in its January 2007 meeting in New Zealand. Pending final resolution of that matter, NIST has adopted the format defined in Table 4 and Table 5. Table 4 is a modified version of Table 7 in ISO/IEC 19794-2, and it differs in

- that the record header shall be absent,
- that the view header shall be present,
- the organization of the compact card minutiae data is different from the record format, and
- the extended data block length shall be absent.

PC-based implementations shall produce templates in the form to Table 4 (lines 12-20, unshaded). MOC implementations shall accept templates in the TLV format of Table 5.

Table 4 – NIST profile of ISO/IEC 19794-2 profile

#	Field	Size (bits)	Valid Values	Notes	NIST Requirement for Evaluation
1.	Format Identifier	32	0x464D5200	"FMR " – finger minutiae record	Record Header The record header on lines 1-11 shall be absent
2.	Version of this standard	32	n n n 0x0	" XX", with XX = 20 or greater	
3.	Length of total record bytes	32	24 to 2^32	from 0x0018 to 0x0000FFFFFFFF	
4.	Capture Equipment Certification	4		compliance with Annex B or future ISO standards	
5.	Capture Device Type ID	12		vendor specified	
6.	Image Size in X	16		in pixels	
7.	Image Size in Y	16		in pixels	
8.	X (horizontal) Resolution	16		in pixels per cm	
9.	Y (vertical) Resolution	16		in pixels per cm	
10.	Number of Finger Views	8	1		
11.	Reserved byte	8	00	RFU, 0 for this version	
12.	Finger Position	8	0 to 10	See ISO/IEC 19794-2 Table 2	View header The view header on
13.	View Number	4	0	Only one view	

14.	Impression Type	4	0 to 3 or 8	See ISO/IEC 19794-2 Table 3	
15.	Finger Quality	8	0 to 100	0 to 100	
16.	Number of Minutiae	8	[0,128]	Denoted by S	
17.	X coordinate	8	[0,255]	Expressed in units of 0.1 mm	View data S instances of the minutiae data on lines 17-20 shall be present
18.	Y coordinate	8	[0,255]	Expressed in units of 0.1 mm	
19.	Minutiae type	2			
20.	Minutiae angle	6	[0,63]	Resolution is 5.625 degrees	
21.	Extended Data Block Length	16	0	0x0000 = no private area	Absent
Remaining structure of extended data block is not presented here.					

Table 5 – ISO/IEC 19794-2 minutiae template DO

Tag	L	Value					
'7F2E'	L1	Biometric data template					
		Tag	L	Value			
		'90'	L2	Finger minutiae data			
				Field	Size (bits)	Valid Values	
				Finger Position	8	0 to 10	One instance of this view header
				View Number	4	0	
				Impression Type	4	0 to 3 or 8	
				Finger Quality	8	0 to 100	
				Number of Minutiae	8	[0,128]	
				X coordinate	8	[0,255]	S instances
				Y coordinate	8	[0,255]	
				Minutiae type	2		
				Minutiae angle	6	[0,63]	

A non-ISO requirement is for the minutia points to be unique. '90' is INCORRECT HERE. INSTEAD SEE 7816-11 Table C.8 – MAY 24: INTENTION IS TO DELETE 4 BYTE HEADER ABOVE.

6.2. Use of header

The four-byte view header on lines 12 to 16 of Table 4 is mandated here because the matcher needs to know (at least) the actual number of minutiae being supplied. The matcher may also find the finger position and quality useful.

6.3. Record length

When S minutiae are present (value on line 16), the length of the binary ISO/IEC 19794-2 compact card data is 4 + 3S bytes. Within a complete APDU the TLV lengths are, for single-byte length fields, L1 = 2 + L2 and L2 = 4 + 3S.

6.4. Failure to acquire

Template generators in MINEX evaluations must always produce a template, even if no minutiae are found. Such a template will have length 4 bytes, and the value in the "number of minutiae" field shall be 0. NIST

considers this a valid template. It is useful for correct accounting of template generator failure (see MINEX NIST IR 7296 section 5.3).

7. MOC interface specification

HANGING The following five subsections define the mechanism by which ALL cards shall be accessed for this test. This includes selection of the application, reading and use of the Biometric Information Template (BIT), installation of a reference template, verification, and recovery of a similarity score.

The template generation function, including use of the BIT, is conducted offline.

7.0 Establish Communications

Do an Answer-to-Reset to determine (T=0, T=1 or T=CL)

7.1. Selection of the test application

This card shall be supplied with a dedicated NIST MOC testing application. It shall be invoked once by using the SELECT command in Table 6. The response shall be as in Table 7.

Table 6 – Command APDU for selection of MOC application

Command Parameter	Meaning	Required Value	Note
CLA INS P1-P2	'00' 'A4' = SELECT '04 00'		
L_c field	Length of AID	16	
Data field	AID	'F0 4E 49 53 54 20 4D 4F 43 20 54 53 54 20 50 31'	In ASCII, "≡NIST MOC TST P1 " where P1 connotes Phase 1
L_e field	Empty		

Table 7 – Response APDU from selection of MOC application

Response Parameter	Meaning
Data field	Empty MAY 23 2007 ABANDON IDEA TO RETURN IDS HERE. INSTEAD SEE NEW SECTIONS 7.6 and 7.7. DELETE: Concatenation of two IDs: first for the card, then the CBEFF ID of the matcher CEBFF ID for the matcher, see for example http://fingerprint.nist.gov/minex/QPL.html
SW1-SW2	See ISO/IEC 7816-4

7.2. Store enrollment template on the card

The APDU for replacing the template on the card is shown in Table 8. It uses the PUT DATA instruction to overwrite the existing reference template. DELETE: NIST has not specified the CHANGE REFERENCE DATA command because it is not part of ISO/IEC 7816-4, and because template adaptation techniques are out-of-scope (see section 7.5.2).

Table 8 – Command APDU for storage of reference template

Command Parameter	Meaning
CLA INS P1-P2	'00' 'DB' = PUT DATA '3F FF' = Store anywhere in the current Dedicated File (Application DF)
L_c field	Length of command data field
Data field	Data Object in BER-TLV format to be stored (tag '7F 2E') Identical to Table 5
L_e field	Empty

Table 9 – Response APDU from storage of reference template

Response Parameter	Meaning
Data field	Empty
SW1-SW2	See ISO/IEC 7816-4

7.3. NIST read of the BIT

HANGING NIST will use the command of Table 10 to retrieve the BIT group template of Table 12 per the response of Table 11. NIST seeks to provide asymmetric enrollment and verification templates. This supports, for example, more minutiae in the reference template than in the verification template. However, NIST will not implement this specialization during the image-processing template generation phase (i.e. the API still only contains a generic create_template function call, which does not take a "purpose" flag). Instead this will occur during a pre-match post-processing of the template.

Table 10 – Command APDU for retrieval of biometric information template

Command Parameter	Meaning
CLA INS P1-P2	'00' 'CB' = GET DATA '3F FF' = Retrieve from anywhere in the current Dedicated File (Application DF)
L_c field	'04'
Data field	'5C' '02' '7F 61' Data Object identifier to be retrieved (group of BIT)
L_e field	'00'

Table 11 – Response APDU from retrieval of biometric information template

Response Parameter	Meaning
Data field	Biometric Information Template (see Table 12)
SW1-SW2	See ISO/IEC 7816-4

Table 9 therefore contains two BITs, the first for enrollment and the second for verification templates. These shall be grouped together as a BIT group template. NIST will read this from each submitted card, and store it. NIST may conduct this operation only once, but will use the BITs to parameterize all conversion operations prior to sending to the card.

All instances of a submitted card must have the same BITs. The BIT must contain the data as described in Table 12.

Table 12 - ISO/IEC 19794-2 Biometric Information Template

Tag	Len.	Value									NIST
‘7F61’	Var.	BIT group template									Requirements
		Tag	Len.	Value							
		'02'	1	Number of BITs in the group							2
		‘7F60’	Var.	Biometric Information Template (BIT)							
				Tag	Len.	Value					
				'80'	1	Algorithm reference					01 = ISO-CC 02 = ISO-CN 03 = ISO-REC 04 = INCITS 378
				'83'	1	Reference data qualifier					
				‘A1’	Var.	Biometric Header Template (BHT)					
						Tag	Len.	Value			
						‘87’	2	CBEFF BDB format owner			0101 i.e. JTC1/SC37
						‘88’	2	CBEFF BDB format type			0005 see sec. 7.3.1
						'B1'	Var.	Biometric matching algorithm parameters ISO/IEC 19794-2 Table 14			
								Tag	Len.	Value	
		NEST-ING OK?						'81'	2	Min. and max. numbers of minutiae, see ISO/IEC 19794-2 (subclause 8.3.3, Table 10)	See sec. 7.4.1
								'82'	1	Minutiae order, see ISO/IEC 19794-2 (subclause 8.3.4 and Table 11 and 12)[5]	Native, see sec. 7.4.3
								'83'	1	Feature handling indicator, see ISO/IEC 19794-2 (Table 15)	00000000b and see sec. 7.4.4
		‘7F60’	Var.	Biometric Information Template (BIT)							
				Tag	Len.	Value					
				'80'	1	Algorithm reference					
				'83'	1	Reference data qualifier					
				‘A1’	Var.	Biometric Header Template (BHT)					
						Tag	Len.	Value			
						‘87’	2	CBEFF BDB format owner			0101 i.e. JTC1/SC37
						‘88’	2	CBEFF BDB format type			0005 see sec. 7.3.1
						'B1'	Var.	Biometric matching algorithm parameters ISO/IEC 19794-2 Table 14			
								Tag	Len.	Value	
								'81'	2	Min. and max. numbers of minutiae, see ISO/IEC 19794-2 (subclause 8.3.3, Table 10)	See sec. 7.4.1
								'82'	1	Minutiae order, see ISO/IEC	Native, see sec.

[5] The text in this line is a corrected version of that in ISO/IEC 19794-2:2005 Table 14 second-to-last line which should reference subclause "8.3.4" not "8.33".

										19794-2 (subclause 8.3.4 and Table 11 and 12)	7.4.3
								'83'	1	Feature handling indicator, see ISO/IEC 19794-2 (Table 15)	00000000b and see sec. 7.4.4

7.3.1. Variants of the ISO/IEC 19794-2 standard

Clause 9 of the ISO/IEC 19794-2:2005 standard gives the "format type" codes for six variants which differ in the encoding (as in Table 21) and placement requirements on minutiae. Placement variation, such as whether a ridge ending is encoded as the ridge skeleton end-point or as the valley bifurcation, remains an open issue in minutiae interoperability. For the current test, NIST will maintain its MINEX requirement of the latter definition. Thus:

- cards must return a value of 0005 for the "format type" in the BIT, and
- encoders should follow the ISO/IEC 19794-2 clause 6 guidance on placement.

7.4. NIST use of the BIT

HANGING Tables 1 and 2 of ISO/IEC 7816-11 define the BIT and grouping structure shown in Table 12. This will be used to parameterize the production of the verification template prior to it being sent to the card. The following sections describe how.

Note that NIST assumes that any requirement by a PC-based matcher to do BIT-like parameterization of its input will occur internally to the matcher. NIST will therefore not support BIT parameterization functionality for off-card matching.

7.4.1. Number of minutiae

NIST's considers that its role is not to impose algorithmic constraints. We therefore impose no limit on the minimum and maximum numbers of minutia a card may request except as follows

- The one-byte value implies a range of [0,255],
- Because some templates will naturally contain 0 minutia (see FTA section 6.4), minimum values may be ignored,
- NIST imposed a 128 minutia maximum in MINEX. This is arguably too high, given that the MINEX 04 trials, using four large operational single index finger flat-impression datasets, found that the leading systems produced a median of 41 minutiae from each image with the 5% and 95% quantiles being 24 and 61 respectively.
- A T=0 APDU command constrains the maximum number of minutia to 60. Note that T = 0 is not required by NIST.
- Informative Annex D.1.1 of ISO/IEC 19794-2 recommends the minimum number of minutiae for enrollment to be 16, and for verification, 12. It also recommends the maximum number of minutiae for enrollment and verification is 60. NIST
  - notes that these are recommendations only,
  - takes no immediate position on the appropriateness of these numbers, and
  - offers section 8 as informational material to suppliers.

In a verification attempt, NIST will send single-view templates to the card for matching. If,

- the value specified in the BIT for the minimum number of minutiae is $0 \le N \le 255$,
- the value specified in the BIT for the maximum number of minutiae is $0 \le M \le 255$,
- the number of minutia present in a candidate, generally third-party, verification template is K, and
- the number of minutia NIST will send to the card is denoted by S then

$$S = \begin{cases} M & \text{if } K \geq M \\ K & \text{if } K < M \\ K & \text{if } K < N \end{cases}$$

Note that N is ignored. This is necessary because some input templates will have zero minutiae. The matcher must execute successfully in such situations.

NIST will reject cards for which N > M.

7.4.2. Pruning mechanism

Operationally a request for N minutiae would be sent to the minutiae extractor. However, such specialization in the context of an offline NIST evaluation involving T templates, N template generators and C cards, imposes the requirement to execute O(TNC) image-to-template conversions. This is likely to be computationally prohibitive, and therefore NIST proposes to standardize a minutiae template reduction process as follows.

The mechanism for pruning minutiae from an input template will be to apply a polar distance reduction strategy based on the ISO/IEC 19794-2 guidance given for polar ordering (subclause 8.3.4). That is, our software will:

- remove K - M minutiae for which the integer quantity

$$r^2 = (x - x_c)^2 + (y - y_c)^2$$

  is largest, while following the ISO guidance on retention of small angle minutiae in the case of r^2 ties. The center of mass (x_c, y_c) will be computed using all K initial minutiae, per the ISO guidance.
- not alter the order of the input minutiae.

Note that archival imagery used in all planned MINEX evaluations is at most 500 pixels in width and height, and is scanned at 19.7 pixels mm^{-1}, and therefore all possible minutiae coordinates can be encoded in 8 bits without sorting (or removal). NIST will publish open-source "C" code in due course.

7.4.3. Sort order of minutiae

Although template generators produce templates whose minutiae are ordered arbitrarily, the ISO-CC standard defines several geometric orderings of the minutia. The x-y and y-x sorting methods support extension of the spatial range of a fingerprint (e.g. for rolled prints) in one dimension. The polar method supports a center-first sort.

Currently NIST intends to support at least the unsorted, Cartesian y-x, Cartesian x-y and polar sorting methods, because the standard defines these as options. NIST will publish open-source "C" code in due course. NIST does not intend to accept commercial code for this purpose, although we may institute a conformance test for implementations that do.

However NIST is aware that commercial readers will need to include such software in addition to the pruning software. This adds complexity and a "degree of freedom" that would better be handled as a natural property of the matching algorithm. Although NIST notes the European Citizen Card specification, CEN/TS 15480-2, requires implementations to accept arbitrarily sorted data, NIST prefers not to ignore the SC37/WG3 intent to allow sorting. NIST does consider that the exact requirements of 19794-2 clause 8 are not clear.

7.4.4. Ridge count, core and delta information

The ISO/IEC 19794-2 standard allows the BIT to request the "extended data" defined in clause 7.5 of that standard. However in Table 12, the binary value 0000000b indicates that ridge count, core and delta information is out of the scope of this test. Suppliers must adhere to the zero specification here.

This implies, in addition, that fully proprietary data is prohibited also.

NIST is interested in possible performance improvement associated with the use of richer templates, but past experience (MINEX 04) has not suggested large improvements are available. However NIST will, on request from any interested party, consider conducting tests of templates that include extended data.

7.5. Verification

HANGING The verification data is sent using the VERIFY command:

Table 13 – Command APDU for comparison of biometric templates

Command Parameter	Meaning
CLA INS P1-P2	'00' '21' = VERIFY '00 00'
L_c field	Length of command data field
Data field	Identical to Table 5 '7F 2E' ISO/IEC 19794-2 minutia template 'xx' Length of template Value Field of the template as described in table 5
L_e field	'00'

Table 14 – Response APDU from comparison of biometric templates

Response Parameter	Meaning
Data field	Empty
SW1-SW2	'90 00' (yes) or '63 C0' (register 0) or, '63 00' or '63 LL' (info available)

NOTE that this only returns the status code. The required similarity score is returned in a separate GET DATA command, see section 7.5.1.

Cards shall remain operable independent of prior verification decisions (i.e. it shall not lock after three no decisions).

7.5.1. Similarity scores

NIST must be able to read a similarity score from the card. NIST will not evaluate cards that produce only a verification decision. These requirements support computation of a full DET characteristic, the primary output of this test. Matcher providers are cautioned that NIST considers matching algorithms that produce only a small number of possible similarity values (naturally or otherwise) to be operationally less useful.

Table 16 specifies return of a two-byte similarity score. Native matching scores outside the range [0,65535] should be remapped by the application.

Table 15 – Command APDU for retrieval of verification similarity score

Command Parameter	Meaning
CLA	'00'

INS P1-P2	'CB' = GET DATA '3F FF' = Retrieve from anywhere in the current Dedicated File (Application DF)
L_c field	'03'
Data field	'5C' '01' 'C0' Data Object identifier to be retrieved (two byte sim score)
L_e field	'04' (2+2) Length of the BER-TLV encoded data object to be retrieved

Table 16 – Response APDU for retrieval of verification similarity score

Response Parameter	Meaning	NIST required values
Data field	Score from the last comparison 'C0' Tag of the score data '02' Length of the score value xx xx Score value	[0-65535] ENDIAN??
SW1-SW2	See ISO/IEC 7816-4	

Note: Using a proprietary tag ('C0') to retrieve the verification similarity score is a guarantee this information will not be available for card applications in operational mode as the information uses a tag which has a meaning for the test application only.

7.5.2. Prohibition of stateful behavior

All components in this test shall be stateless and idempotent. No component of the test is permitted to maintain state information. This applies to template generation and matching, and to on-card and off-card activity. NIST will institute appropriate tests to detect stateful behavior in the activities mentioned in the following subsections.

NOTE NIST is prohibiting template adaptation, and will accordingly implement checks to detect any stateful behavior and side-effects. NIST will cease evaluation and inform the provider. However NIST is interested in template update as a potential means of improving operational performance. If there is demand NIST may conduct a dedicated evaluation of the technology in a future test.

7.6. Reading card identifier

Is ISO/IEC 7816-4:2004 clause 8.1.1.2.6 useful here?

Table 17 – Command APDU for retrieval of Card identifier

Command Parameter	Meaning
CLA INS P1-P2	'00' 'CB' = GET DATA '3F FF' = Retrieve from anywhere in the current Dedicated File (Application DF)
L_c field	'03' length of command data field
Data field	'5C' '01' '66' Data Object identifier to be retrieved (Card Data)
L_e field	'00'

The response field should contain a discretionary field (tag '73') containing the card version information in tag '88'. The matcher information might also be found in the same response when tag '99' is present in the same discretionary field.

Table 18 – Response APDU for retrieval of Card identifier

Response Parameter	Meaning	NIST required values
Data field		
SW1-SW2	See ISO/IEC 7816-4	

7.7. Reading matcher identifier

Table 19 – Command APDU for retrieval of Matcher identifier

Command Parameter	Meaning
CLA INS P1-P2	'00' 'CB' = GET DATA '3F FF' = Retrieve from anywhere in the current Dedicated File (Application DF)
L_c field	'03' length of command data field
Data field	'5C' '01' '6E' Data Object identifier to be retrieved (Card Data)
L_e field	'00'

The response field should contain a discretionary field (tag '73') containing the matcher information in tag '99'.

Table 20 – Response APDU for retrieval of Matcher identifier

Response Parameter	Meaning	NIST required values
Data field		
SW1-SW2	See ISO/IEC 7816-4	

8. Comparison of the INCITS 378 and ISO/IEC 19794-2 standards

CBEFF Format Owner = 0101 for ISO/IEC JTC 1/SC 37.

CBEFF Format Type = 0005 for ridge endings encoded as valley bifurcation points, as in INCITS 378.

The INCITS 378 standard's record format and the ISO/IEC 19794-2 compact card format differ syntactically and semantically. These differences are presented in the next two subsections. Thereafter we address the implications for transcoding between the standards, and the MINEX requirements.

8.1. Syntactic differences

The INCITS 378:2004 and ISO-CC templates differ as follows. In ISO-CC

- the (x,y) coordinates are encoded in 8 bits as opposed to 14 bits,
- the spatial resolution is fixed at 10 pixels per millimeter as opposed to variable resolution,
- the angle is encoded in 6 bits as opposed to 8 bits, and
- the minutia quality value is absent.

These differences are depicted in Table 21. Note that ridge count and core and delta information is not being evaluated in this test, and therefore the BIT entry defined in ISO/IEC 19794-2 Table 14, shall be set to 0, and ignored.

Table 21 – Minutia encodings of the ISO-CC and INCITS 378

	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
ISO/IEC 19794-2 Compact Card Format	x coord.	y coord.	t angle			
	8	8	2 6			
ISO/IEC 19794-2 Record Format and INCITS 378	t x coord.		r y coord		angle	quality
	2 14		2 14		8	8
	t = type		r = reserved			

8.2. Semantic differences

The abbreviated 8 bit (x,y) encodings in the ISO-CC standard support "typical" single-finger images by specifying a hard-wired resolution of 10 pixels per millimeter (ISO/IEC 19794-2, subclause 8.2). This is approximately half of typical enrollment data gathered on 500 ppi (19.7 pixels per millimeter) optical scanners. This is summarized in Table 22. The effect would be that minutia can not extend over a region larger than 255 / 10 = 25.5 mm. However the ISO-CC standard provides for coordinate wraparound wherein the minutia coordinates are sorted such that the actual value may extend beyond the range by encoding it as x mod 256. Reconstruction of the actual value is possible because sorting is applied.

Table 22 – Minutia location quantization of ISO-CC and INCITS 378 templates

Standard	Allowed values (units)	Allowed values (mm)	Allowed values (mm), at 500 ppi
INCITS 378:2004 record format	[0,16383]	Depends on the encoding resolution	[0, 0.051, 0.1015, 0.1523, …, 831.6]
ISO/IEC 19794-2 record format			
ISO/IEC 19794-2 compact card	[0,255]	[0, 0.1, 0.2 … 25.5]	N/A

As shown in Table 23, the three different angular encodings support minutia encodings of varying precision. Whether this difference materially affects performance is dependent on the sensitivity of the matching algorithm, and on how accurately template generators measure the angle.

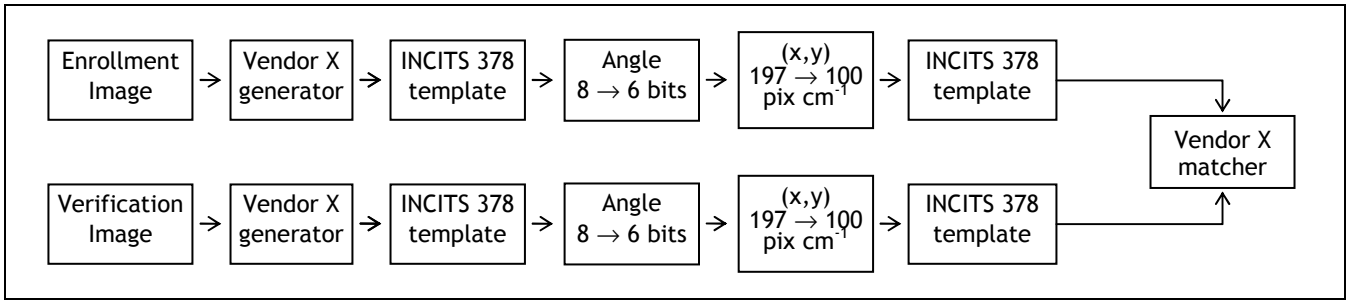
Table 23 – Minutia angle quantization of ISO-CC and INCITS 378 templates

Standard	Allowed values (units)	Quantization	
		Degrees per unit	Radians per unit
INCITS 378:2004 record	[0,179]	360 / 180 = 2	2pi / 180 = 0.0349
ISO/IEC 19794-2 record	[0,255]	360 / 256 = 1.4063	2pi / 256 = 0.0245
ISO/IEC 19794-2 compact card	[0,63]	360 / 64 = 5.625	2pi / 64 = 0.0982

8.3. Simulation protocol

To quantify the effects of the ISO-CC encoding vs. the INCITS 378 record format, NIST simulated the production of ISO-CC templates by quantizing the coordinates and angles in sets of INCITS 378 templates produced in NIST's existing MINEX testing.

Figure 2 - Use of INCITS 378 for simulation of ISO-CC accuracy



The C code fragments to effect these transformations are shown in Table 24. This code is part of the "fmrisocompact" program distributed[6] by NIST as part of a suite of libraries and applications that support various INCITS biometric standards.

Table 24 – Source code for conversion of INCITS 378 to ISO-CC

Angular re-quantization

```
// The ISO minutia record has 6 bits for the angle, so
// we have 64 possible values to represent 360 degrees.
for (m = 0; m < mcount; m++)                                          // for all minutiae
{
   const int theta = 2 * (int)fmds[m]->angle;                         // 378 has 2 degrees
   const double isotheta = round((64.0 / 360.0) * (double)theta);     // CC has 5.625 deg
   fmds[m]->angle = (unsigned char)round(((360.0 / 64.0) * isotheta) / 2.0); // Put back in 378
}
```

Spatial re-quantization

```
for (m = 0; m < mcount; m++)                        // for all minutiae
{
   const double x = (double)fmds[m]->x_coord;       // cast from unsigned short
   const double y = (double)fmds[m]->y_coord;       // cast from unsigned short

   const double xmm = 10.0 * x / (double)xres;      // millimeters, because INCITS 378 resolution
   const double ymm = 10.0 * y / (double)yres;      // values are in pixels per centimeter
   const double xunits = xmm / 0.1;                 // units of 0.1 pix per mm which is the CC
   const double yunits = ymm / 0.1;                 // card format's hardwired sampling freq
   const unsigned short xcc =
            (unsigned short)(0.5 + xunits);         // round the value - this is what would be
   const unsigned short ycc =                       // stored in "typical" say 500 dpi operation
            (unsigned short)(0.5 + yunits);

   fmds[m]->x_coord = (unsigned short)((double)xcc * (double)xres * 0.01);  // Now put back
   fmds[m]->y_coord = (unsigned short)((double)ycc * (double)yres * 0.01);  // in 378 format
}
```

We took five vendors and applied their matchers to INCITS 378 templates derived from the outputs of that vendor's template generator. They are applied natively, that is to enrollment and verification templates from the same supplier's INCITS 378 generator. The five matching algorithms, identified in the first columns, are some of the more accurate ones assessed in the Ongoing MINEX evaluation.

All results apply to single finger matching using the POEBVA collection of left and right index fingers, as used in the Ongoing MINEX evaluation. The FNMR computation is conducted over 15 disjoint sets each containing 16000 genuine scores. The FMR computation is likewise conducted over 15 disjoint sets each containing 16000 impostor scores.

8.4. Results after re-quantization of angle and position

Table 25 gives the change in FNMR when the matcher operating threshold is set to achieve FMR values of 0.01 on INCITS 378 templates and then reset to achieve those FMRs on ISO-CC templates. This thresholding

[6] This is available for download; see http://www.itl.nist.gov/iad/894.03/nigos/incits.html

strategy is representative of the situation where thresholds can be set for the ISO-CC template independently of any INCITS 378 matcher trial or calibration.

Table 25 – Relative accuracy of ISO-CC and INCITS 378 templates (Fixed FMR)

	Base FMR	Base FNMR	Change in FNMR	
A1	0.01	0.0140	0.0005 +/- 0.0006	p 2e-03
A2	0.01	0.0158	0.0018 +/- 0.0007	p 3e-08
A3	0.01	0.0133	0.0007 +/- 0.0007	p 2e-03
A4	0.01	0.0183	0.0006 +/- 0.0005	p 6e-05
A5	0.01	0.0159	0.0008 +/- 0.0005	p 9e-05

Table 26 gives changes in FMR and FNMR when the matcher operating threshold is set to achieve FMR values of 0.01, on the unaltered INCITS 378 templates. This thresholding strategy is representative of the situation where a ISO-CC template is sent, with transcoding, to an INCITS 378 matcher.

Table 26 – Relative accuracy of ISO-CC and INCITS 378 templates (fixed t)

Change in FMR and FNMR when ISO-CC encoding is synthesized from instances of the INCITS 378 record format "MIN:A" templates.					
	Base FMR	Change in FMR		Base FNMR	Change in FNMR	
A1	0.01	-0.0013 +/- 0.0008	p 7e-06	0.0140	0.0008 +/- 0.0008	p 5e-04
A2	0.01	-0.0008 +/- 0.0011	p 1e-02	0.0158	0.0020 +/- 0.0007	p 4e-09
A3	0.01	-0.0006 +/- 0.0008	p 6e-03	0.0133	0.0007 +/- 0.0006	p 5e-04
A4	0.01	-0.0004 +/- 0.0006	p 1e-02	0.0183	0.0007 +/- 0.0007	p 7e-04
A5	0.01	-0.0008 +/- 0.0009	p 1e-03	0.0159	0.0010 +/- 0.0008	p 1e-04

Thus when going from INCITS 378 to ISO-CC

- Table 25 shows small but statistically significant increases in FNMR. The worst case is A2 whose FNMR goes from 0.0158 to 0.0176 at a FMR of 0.01 corresponding to about 11% more missed matches.
- At a fixed threshold for both kinds of template, Table 26 shows small that FMR is slightly but significantly lower for ISO-CC than for INCITS 378, but that FNMR is again higher.

The presence of an algorithm-effect (some implementations are more sensitive than others, viz. A2 over A3, means that vendors should consider this issue for their implementations.

NOTE The change in performance for less accurate matchers and template generators has not been studied.

8.5. Results after reduction in number of minutiae

The plots of Figure 3 show the effect of apply the pruning operation of section 7.4.2 to raw INCITS 378 (not ISO-CC) templates. The effect on accuracy is shown for the same five leading matcher providers as used above. The matchers are applied natively i.e. the matcher from provider X is applied to compare templates from X's generator. For each matcher the threshold is set to the value that gives a FMR of 0.001 on the entire unpruned corpus. The error rates are then recomputed after removing zero or more minutia to achieve a maximum of N in each template. The top graph shows the effect of retaining all minutiae in the enrollment template and minutiae in the verification template. The lower graph shows the effect of pruning both.

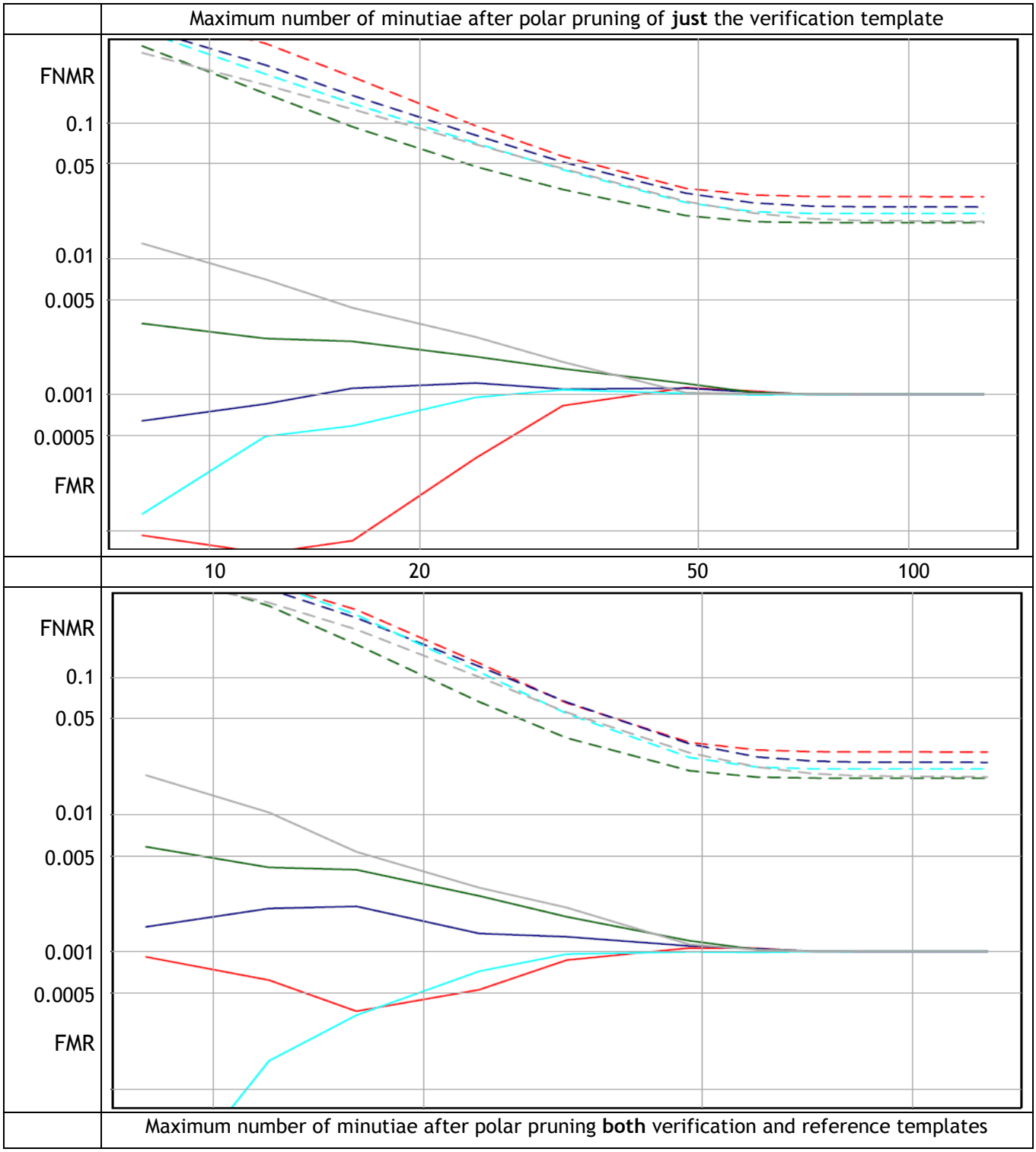
The conclusions are

- FNMR increases for all matchers algorithms, with approximately 60 minutia being a reasonably "lossless" value. Note that the 95% percentiles for number of minutiae are 60, 65, 60, 64, and 63.
- FNMR has increased by an order of magnitude (from ~2% to ~20%) when fewer than 20 minutia are used.

- FMR values also change significantly, beginning at the same 60 minutiae breakpoint, and substantially below about 20 minutiae. Some systems improve (decline in FMR) and some degrade (increase in FMR). NIST is concerned that any increase in FMR associated with transactions involving small numbers of minutiae is a security hazard.

Not shown here is distributional information on the numbers of minutiae produced by template generators. NIST has observed variation in the numbers found from a single image. Also not shown are interoperable results (cross-vendor) and NIST suggests that an incorrect conclusion from the graphs would be that template generators finding more minutiae are better performing.

Figure 3 – Effect of minutia pruning on FNMR (above) and FMR (below)



8.6. Conclusions

The ISO-CC template can offer performance approaching that of the INCITS 378 template. However, some implementations exhibit degraded performance. This study, however, only approximates actual ISO-CC

performance because providers may be able to improve algorithmic functionality if they specifically know the target result is ISO-CC.

9. Privacy requirements

Move clause 9 into section 11 or 12

9.1. Returning software to vendors

NIST will not return any supplied software, documentation, or other material to vendors.

9.2. Returning cards to vendors

NIST will not generally return cards to the provider. NIST will destroy the cards within 90 days of publication of the results for that card or notification to the vendor that the card is inoperable. This requirement is needed because NIST must maintain the privacy of templates we may send to the card. This arises because our test specification does not include a mechanism for the purging of templates from the card (null overwrite may not be sufficient). However, NIST will return cards during an initial acceptance testing phase, if errors are encountered. This will support debugging and resolution.

10. Ambiguities in the standards

The first (hanging) paragraph of clause 8 of ISO/IEC 19794-2 reads as follows (emphasis is ours)

> This standard defines two card related encoding formats for finger minutiae, the normal size format and the compact size format. Such a format may be used e.g. as part of a Biometric Information Template as specified in ISO/IEC 7816-11 with incorporated CBEFF data objects, if off-card matching is applied, or in the command data field of a VERIFY command, if match-on-card (MOC) is applied (see ISO/IEC 7816-4 and -11).

This text does not indicate what format the enrollment template should be in. If it is considered to be proprietary then standard should discuss the issue (in a NOTE).

Clause 6.6 in 19794-2, on how to match typed-minutia, is overly prescriptive, and should be deleted.

The text in ISO/IEC 19794-2:2005 Table 14 second-to-last line references subclause "8.33" which should be "8.3.4".

The last line of clause 8.3.4 "The same construction principle may be applied also for the Y coordinate" should be changed to "The same construction principle may alternatively be applied to the Y coordinate. Using this construction on X and Y together is not possible."

There should be a normative requirement for minutia (x,y,theta) triples to be unique – but not for (x,y) only.

11. API specification

11.1. Overview

This entire section 11 is essentially a cut-and-paste of the Ongoing MINEX API. It is subject to great change and is not the definitive API for the MINEX II test. It is included here ONLY has an indication of what the PC-side API will look like. It does not include any card-related material. It is, however, open for comments.

The Minutiae Interoperability Exchange Test (MINEX) is an ongoing program to measure the performance of fingerprint matching software utilizing interoperable minutiae-based fingerprint templates. The content and format of those interoperable minutiae-based fingerprint templates are defined in this specification and are hereafter referred to as MINEX compliant templates.

Those wishing to submit software for MINEX testing shall be required to provide NIST with an SDK (Software Development Kit) library which complies with the API (Application Programmer Interface) specified in this document. At a minimum, the SDK submitted must provide functionality to create MINEX compliant templates based on individual fingerprint images. Support for matching pairs of MINEX compliant templates is encouraged, but optional.

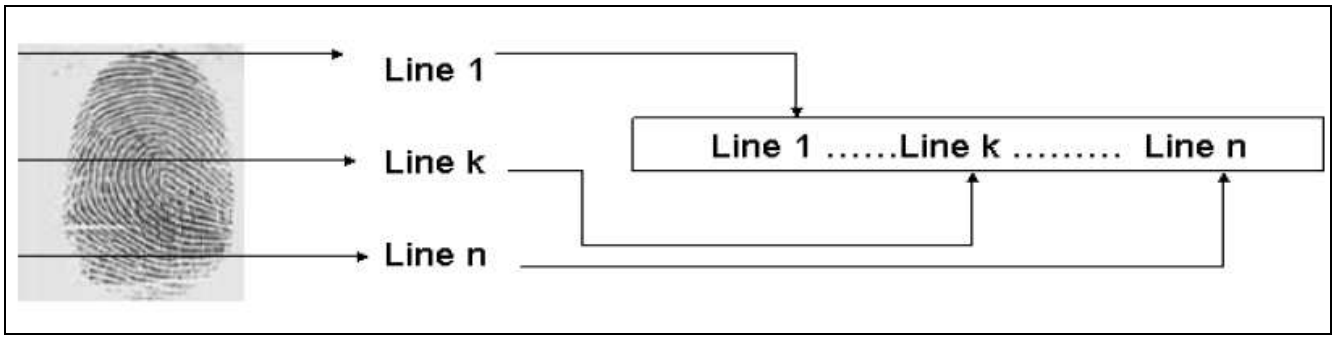
In addition to providing a general platform for testing the performance of interoperable fingerprint systems, MINEX provides a mechanism for testing compliance with NIST Special Publication 800-76 [1] (refer to section 7.4.1).

11.2. Fingerprint image data

11.2.1. Format

The SDK must be capable of processing fingerprint images supplied to the SDK in uncompressed raw 8-bit (one byte per pixel) grayscale format. Each image shall appear to have been captured in an upright position and approximately centered horizontally in the field of view. The image data shall appear to be the result of a scanning of a conventional inked impression of a fingerprint. Figure 1 illustrates the recording order for the scanned image. The origin is the upper left corner of the image. The x-coordinate (horizontal) position shall increase positively from the origin to the right side of the image. The y-coordinate (vertical) position shall increase positively from the origin to the bottom of the image.

Figure 4 – Fingerprint image raster scan order



Raw 8-bit grayscale images are canonically encoded. The minimum value that will be assigned to a "black" pixel is zero. The maximum value that will be assigned to a "white" pixel is 255. Intermediate gray levels will have assigned values of 1- 254. The pixels are stored left to right, top to bottom, with one 8-bit byte per pixel. The number of bytes in an image is equal to its height multiplied by its width as measured in pixels; there is no header. The image height and width in pixels will be supplied to the SDK as supplemental information.

11.2.2. Resolution and dimensions

All images for this test will employ 500 PPI resolution (horizontal and vertical).

The dimensions of the fingerprint images will vary from 150 to 812 pixels in width, and 166 to 1000 pixels in height.

Note – the SDK must be capable of processing images with any dimensions in these specified ranges without the use of separately invoked cropping or padding facilities. For example, SDKs which require cropping of large images must do so internal to the operation of the create_template (see below) API call.

11.2.3. Sensor and impression types

All images used for testing in MINEX come from the POEBVA data set described in NISTIR 7296 [2] (see Appendix B, Table 23 page 47) and thus have been obtained from live-scan sensors (Smiths-Heimann ACCO 1394 and Cross Match 300A). All images tested in MINEX are plain impression type images.

11.3. INCITS 800-76 compliant templates

To be considered MINEX compliant templates, all templates created must be compliant with NIST Special Publication 800-76 [1] (refer to Table 12, page 26). Two additional constraints imposed by MINEX upon the template requirements defined above are:

- The Minutiae Quality field for each minutia shall be set to 0.
- The Finger Quality field will be input by the test application and shall be output identically by the SDK at run-time. (I.e., the SDK is not to generate this value)
- Past participants in MINEX04 [2] may note that the requirements for templates specified by the Ongoing MINEX test are identical except for the fields listed below:
- In MINEX04, the field Finger Quality field had a range of values resulting from re-mapping the NIST NFIQ [3] quality values (1 through 5) to the values 100,75,50,25 and 1 respectively. However, 800-76 re-maps these same NFIQ quality values to 100,80,60,40, & 20 respectively.
- In MINEX04, the field Impression Type had a range of 0 through 3. However, 800-76 limits the range of values to 0 and 2.

11.4. Testing interface description

MINEX participants shall submit an SDK which provides the following interface (shown in C-style pseudo-code prototypes).

11.4.1. Pre-defined values

The pre-defined values (constants) of Table 27 are for use in specifying parameters to the MINEX testing interface:

Table 27 – MINEX predefined values

```
// Finger quality values
#define QUAL_POOR 20 // NFIQ value 5
#define QUAL_FAIR 40 // NFIQ value 4
#define QUAL_GOOD 60 // NFIQ value 3
#define QUAL_VGOOD 80 // NFIQ value 2
#define QUAL_EXCELLENT 100 // NFIQ value 1
```

```
// Impression type codes
#define IMPTYPE_LP 0x00 // Live-scan plain
#define IMPTYPE_NP 0x02 // Nonlive-scan plain
```

```
// Finger position codes
#define FINGPOS_UK 0x00 // Unknown finger
#define FINGPOS_RT 0x01 // Right thumb
#define FINGPOS_RI 0x02 // Right index finger
#define FINGPOS_RM 0x03 // Right middle finger
#define FINGPOS_RR 0x04 // Right ring finger
#define FINGPOS_RL 0x05 // Right little finger
#define FINGPOS_LT 0x06 // Left thumb
#define FINGPOS_LI 0x07 // Left index finger
#define FINGPOS_LM 0x08 // Left middle finger
#define FINGPOS_LR 0x09 // Left ring finger
#define FINGPOS_LL 0x0A // Left little finger
```

11.4.2. Minutiae extraction

NIST will create templates using the function call defined in Table 28.

Table 28 – MINEX create_template API Function

Prototype	INT32 create_template(const BYTE* raw_image, const BYTE finger_quality, const BYTE finger_position, const BYTE impression_type, const UINT16 height, const UINT16 width, BYTE *template);
Description	This function takes a raw image as input and outputs the corresponding MINEX compliant template. The memory for the template is allocated before the call (i.e., create_template() does not handle the memory allocation for the template parameter). The function returns either success (0) or failure (non-zero). Failure indicates a failure to enroll the image and will result in the output of a null template which will be used in later comparisons. Note – null templates are defined as containing the Record header and Finger View header only, with zero minutiae information (i.e. Number of Minutiae shall be set to 0). Thus, it is a 32 byte template (26-byte Record Header + 4-byte Finger View header + 2 bytes for the Extended Data Block length which is 0x0000). All other fields in the Record and Finger View headers shall be set to their regular and accurate values.
Parameters	raw_image (input): The uncompressed raw image used for template creation. finger_quality (input): The quality of the fingerprint image (e.g. QUAL_GOOD). finger_position (input): The finger position code (e.g. FINGPOS_RI). impression_type (input): The impression type code (e.g. IMPTYPE_LP). height (input): The number of pixels indicating the height of the image. width (input): The number of pixels indicating the width of the image. template (output): The processed template.
Return Value	This function returns zero on success or a documented non-zero error code otherwise

11.4.3. Minutiae extraction and matching

One-to-one comparisons representative of single-finger verification attempts will be made using the function defined in Table 29.

Table 29 – MINEX API match_templates function

Prototype	INT32 match_templates(const BYTE *probe_template, const BYTE *gallery_template, float *score);
Description	This function compares two MINEX compliant templates and outputs a match score. The probe_template parameter shall be compared to the gallery_template parameter (in that precise order where the underlying matcher is order dependent). The score returned is a floating-point number which represents the similarity of the original fingerprint images from which the templates where created. Scores should not be quantized. It may be assumed that memory for the score parameter is allocated before the call. Note that comparisons in which either template is a null template (see 3.2.2 above) shall cause the matching operation to fail and output a documented error code (see 3.3 below).
Parameters	probe_template (input): A template returned by create_template(). gallery_template (input): A template returned by create_template(). score (output): A similarity score resulting from comparison of the templates.
Return Value	This function returns zero on success (i.e. a valid score was produced) or a documented non-zero error code on failure. In the latter case, the function shall return a score of -1. Note – If the legitimate range of match scores includes the value -1, the participant must inform the MINEX Test Liaison.

11.4.4. Implementation identifier

The implementation shall support the function of Table 30 to identify itself.

Table 30 – MINEX API get_pids function

Prototype	INT32 get_pids(UINT32* feature_extractor, UINT32* matcher);
Description	This function retrieves CBEFF PID information which identifies the SDK's core feature extractor and (if supported) template matcher. The PID output for feature_extractor shall be identical in both format and value to the CBEFF Product Identifier (PID) defined by INCITS 378-2004 [4] (refer to section 6.4.4). If the SDK supports template matching functionality the PID output for matcher shall have the two most significant bytes (specified by INCITS 378-2004 as identifying the "owner" of the technology) set to values identical to the corresponding bytes of feature_extractor. Otherwise, if the SDK does not support matching functionality, the PID value returned for matcher shall be 0. . It may be assumed that memory for the feature_extractor and matcher parameters are allocated before the call. Note that the two least significant bytes of the CBEFF PID are defined by INCITS 378-2004 as identifying the version of the feature extractor (referred to as PID "Type"). The two least significant bytes of feature_extractor and matcher shall be set as specified by INCITS 378-2004 (i.e. they may either be set to 0, or to a version number assigned by the "owner" of the technology).
Parameters	feature_extractor (output): A PID which identifies the SDK's feature extractor. matcher (output): A PID which identifies the SDK's matcher.
Return Value	This function returns zero on success or a documented non-zero error code on failure. In the latter case, both output parameters shall be set to 0.

11.4.5. Error Codes and Handling

The participant shall provide documentation of all (non-zero) error or warning return codes (see section 4.3, Documentation).

The application should include error/exception handling so that in the case of a fatal error, the return code is still provided to the calling application.

At minimum the return codes of Table 31 shall be used.

Table 31 – MINEX API return codes

Return code	Explanation
0	Success
1	Image size not supported
2	Failed to extract minutiae – unspecified error
3	Failed to extract minutiae – impression type not supported
4	Failed to match templates – null probe or gallery template
5	Failed to match templates – unable to parse probe template
6	Failed to match templates – unable to parse gallery template

All messages which convey errors, warnings or other information shall be suppressed.

11.5. Software and Documentation

11.5.1. SDK Library and Platform Requirements

Individual SDKs provided must not include multiple "modes" of operation, or algorithm variations. No switches or options will be tolerated within one library. For example, the use of 2 different "coders" by a minutiae extractor must be split across 2 separate SDK libraries.

Participants shall provide NIST with binary code only (i.e. no source code) – supporting files such as header (".h") files notwithstanding. It is preferred that the SDK be submitted in the form of a single static library file (ie. ".LIB" for Windows or ".a" for Linux). However, dynamic/shared library files are permitted.

If dynamic/shared library files are submitted, it is preferred that the API interface specified by this document be implemented in a single "core" library file with the base filename 'libminex' (for example, 'libminex.dll' for Windows or 'libminex.so' for Linux). Additional dynamic/shared library files may be submitted that support this "core" library file (i.e. the "core" library file may have dependencies implemented in these other libraries).

Note that dependencies on external dynamic/shared libraries such as compiler-specific development environment libraries are discouraged. If absolutely necessary, external libraries must be provided to NIST upon prior approval by the Test Liaison.

The SDK will be tested in non-interactive "batch" mode (i.e. without terminal support). Thus, the library code provided shall not use any interactive functions such as graphical user interface (GUI) calls, or any other calls which require terminal interaction (e.g. calls to "standard input" or "standard output").

NIST will link the provided library file(s) to a C language test driver application (developed by NIST) using the GCC compiler (for Windows platforms Cygwin/GCC version 3.3.3 will be used; for RedHat Linux 7.3 platforms GCC version 2.96 will be used. All GCC compilers use Libc 6). For example,

```
gcc –o mintest mintest.c -L. –lminex
```

Participants are required to provide their library in a format that is linkable using GCC with the NIST test driver, which is compiled with GCC. All compilation and testing will be performed on x86 platforms running either Windows 2000 or Red Hat Linux 7.3 (dependent upon the operating system requirements of the SDK). Thus, participants are strongly advised to verify library-level compatibility with GCC (on an equivalent platform) prior to submitting their software to NIST to avoid linkage problems later on (e.g. symbol name and calling convention mismatches, incorrect binary file formats, etc.).

11.5.2. Installation and Usage

The SDK must install easily (i.e. one installation step with no participant interaction required) to be tested, and shall be executable on any number of machines without requiring additional machine-specific license control procedures or activation.

The SDK's usage shall be unlimited. No usage controls or limits based on licenses, execution date/time, number of executions, etc. shall be enforced by the SDK.

It is recommended that the SDK be installable using simple file copy methods, and not require the use of a separate installation program. Contact the Test Liaison for prior approval if an installation program is absolutely necessary.

11.5.3. Documentation

Complete documentation of the SDK shall be provided, and shall detail any additional functionality or behavior beyond what is specified in this document.

The documentation must define all error and warning codes.

11.5.4. Speed

On average, a template match operation shall take no more than 10 milliseconds, and a template creation operation shall take no more than 1 second to complete (using a 2GHz Pentium IV).

11.6. References

[1] C. Wilson, et al., "Biometric Data Specification for Personal Identity Verification," NIST Special Publication 800-76-1 http://csrc.nist.gov/publications/nistpubs/800-76-1/SP800-76-1_012407.pdf

[2] P. Grother, et al., "Performance and Interoperability of the INCITS 378 Template," NIST IR 7296 http://fingerprint.nist.gov/minex04/minex_report.pdf

[3] E. Tabassi, et al. "Finger Print Image Quality," NISTIR 7151 2004 (Gaithersburg, MD: National Institute of Standards and Technology, August 2004) http://www.itl.nist.gov/iad/894.03/quality/reports/ir_7151.pdf

[4] American National Standard for Information Technology – Finger Minutiae Format for Data Interchange, ANSI/INCITS 378-2004, www.incits.org

12. Application Form

The remainder of section 12 is a draft application form. It is subject to change and is included here as an indication of what NIST might publish.

Application to Participate in the MINEX II Test. The objectives of this series of tests (MINEX II) are described in the preceding sections of this document.

12.1. Who Should Participate

Developers of ISO/IEC 19794-2 minutia-based systems are invited to participate in MINEX II. In addition, companies, research organizations, or universities that have developed mature prototypes or who research fingerprint matching of interoperable templates are invited to participate.

The fingerprint template generation and matching software need not be "operational," nor a production system, nor commercially available. However, the system must, at a minimum, be a stable implementation capable of being "wrapped" (formatted) in the API specification that National Institute of Standards and Technology (NIST) has specified in section XX for this evaluation.

Anonymous participation will not be permitted. This means that signatories to this Agreement acknowledge that they understand that the results (see Sections UU and VV) of the evaluation of the software will be published with attribution to their organization.

12.2. How to Participate

12.2.1. Application Form

In order to request participation in MINEX II, potential participants must fill out this Agreement, Application to Participate in MINEX II, identifying the Responsible Party and the Points of Contact, print and sign the form, and send to the location designated below.

12.2.2. Parties

The Responsible Party is an individual with the authority to commit the organization to the terms in this document.

The Point of Contact is an individual with detailed knowledge of the system applying for participation.

12.3. NIST Activity

Upon receipt of the signed form by NIST, the organization will be classified as a "Participant". NIST must receive the form by the due date described in the MINEX II Calendar, as posted on the MINEX website (Link).

12.3.1. Supplier validation

Registered Participants should download the small Validation Dataset available on the website. Prior to submission of their Software Development Kit (SDK) the Participant needs to verify that a) their software executes on the validation data, and b) produces a candidate list in the desired format.

The output of the validation data must be submitted to NIST along with the SDK. (NIST will supply the correct match to each search in the Validation Data. However, the submitted validation candidate list need not have the correct match appear on the candidate list.)

12.3.2. Submission of software to NIST

After the Participant has executed his software on the Validation Data, the Participants should send the software (SDK) together with the validation output to NIST. Software must be encrypted using a key provided by NIST. This may be emailed to minex@nist.gov, or sent to NIST on CD media at:

Evaluation of MINEX II Test Liaison
National Institute of Standards and Technology
Information Access Division (894)
100 Bureau Drive, Stop 8940
Gaithersburg, MD 20899-8940

Participants shall submit at least one SDK.

12.3.3. Acceptance testing

Software submitted must be compliant with the MINEX II API Specification, as posted on the MINEX II website at http://fingerprint.nist.gov/minex

Upon receipt of the SDK and validation output, NIST will attempt to reproduce the output by executing the SDK on the validation data, using a NIST computer. In the event of disagreement in the output, or other difficulties, the Participant will be notified.

In the event software is found to be non-functional or non-compliant with the MINEX II API, or where the Validation Dataset results cannot be replicated by NIST, Participants will be notified with a detailed description of the problem(s) and given a reasonable opportunity to resubmit (as time allows) according to the discretion of the MINEX II Liaison.

12.4. Points of Contact

3.1. The MINEX II Liaison is the government point of contact for MINEX II.

3.2. All correspondence should be directed to minex@nist.gov, which will be received by the MINEX II Liaison and other MINEX II personnel.

3.3. These correspondences may be posted on the FAQ (Frequently Asked Questions) area of the MINEX II website at the discretion of the MINEX II Liaison. The identity of those persons or organizations whose correspondences lead to FAQ postings will not be made public in the FAQ.

12.5. Access to MINEX II Validation Data

4.1. The MINEX II Validation Data is supplied to Participants to assist in preparing for MINEX II.

4.2. The fingerprints in the MINEX II Validation Data are representative of the MINEX II Test Data only in format. Image quality, collection device and other characteristics may vary between the Validation and Test Datasets.

12.6. Access to MINEX II Test Data

The MINEX II Test Datasets are protected under the Privacy Act (5 U.S.C. 552a), and will be treated as Sensitive but Unclassified and/or Law Enforcement Sensitive.

MINEX II Participants will have no access to MINEX II Test Data, either before, during or after the test.

12.7. Reporting of Results

12.7.1. Reports

The Government will combine appropriate results into one or more MINEX II Reports. Together these will contain, at a minimum, descriptive information concerning MINEX II, descriptions of each experiment, and aggregate test results. NIST will use DET performance metrics as the primary indicators of one-to-one verification search accuracy. This involves plotting false rejection versus false acceptance rates for all threshold values. NIST will also report enrollment and verification timing information.

NIST may compute and report other aggregate statistics.

12.7.2. Pre-publication Review

Participants will have an opportunity to review and comment on the Reports. Participants' comments will be either incorporated into the main body of the report (if it is decided NIST reported in error) or published as an addendum. Comments will be attributed to the participant.

12.7.3. Citation of the Report

After the release of the Phase II Final Report, Participants may decide to use the results for their own purposes. Such results shall be accompanied by the following phrase: "Results shown from the Minutiae Interoperability Exchange Test (MINEX II) do not constitute endorsement of any particular system by the U. S. Government." Such results shall also be accompanied by the Internet address (URL) of the MINEX II Report on the MINEX II website.

12.7.4. Rights and Ownership of the Data

Any data obtained during MINEX (excepting the submitted SDK itself), as well as any documentation required by the Government from the participants, becomes the property of the Government. Participants will not possess a proprietary interest in the data and/or submitted documentation.

12.8. Agreement to Participate

With the signing of this form, Participants attest that they will not file any MINEX II-related claim against MINEX II Sponsors, Supporters, staff, contractors, or agency of the U.S. Government, or otherwise seek compensation for any equipment, materials, supplies, information, travel, labor and/or other participant provided services.

The Government is not bound or obligated to follow any recommendations that may be submitted by the Participant. The United States Government, or any individual agency, is not bound, nor is it obligated, in any way to give any special consideration to MINEX II Participants on future contracts, grants or other activities.

With the signing of this form, Participants realize that any test details and/or modifications that are provided in the MINEX II website supersede the information on this form.

With the signing of this form, Participants realize that they can withdraw from the MINEX II at any time prior to the start of XXX, without their participation and withdrawal being documented in the MINEX II Final Report.

7.10. NIST will use the Participants SDK software only for the agreed-upon testing, and in the event errors are subsequently found, to rerun prior tests and resolve those errors.

7.11. NIST agrees not to use the Participants software for purposes other than indicated above, without express permission by the Participant.

7.12. Please mail the completed and signed form to:

Via Mail:

MINEX II Liaison
National Institute of Standards and Technology
Information Access Division (894)
100 Bureau Drive, Stop 8940
Gaithersburg, MD 20899-8940

1

To be completed by suppliers of ISO/IEC 7816 cards					
Responsible Party for supplier should complete these boxes					
Company / Organization Name					
Title	First Name	MI	Last Name		Suffix
Street Address					
City		State	Zip	Country	
Phone	Fax		Email		
Point of contact for day to day operations	Phone		Email		
Responsible Party for supplier of smart card should identify the responsible party for a first supplier of any associated software or algorithms present on the card and subject to evaluation, if a different organization. That person must also complete one of these forms, cross designating you.					
Associated Company / Organization Name #1 (write N/A if no partner)					
Title	First Name	MI	Last Name		Suffix
Street Address					
City		State	Zip	Country	
Phone	Fax		Email		
Responsible Party for supplier of smart card should identify the responsible party for a second supplier of any associated software or algorithms present on the card and subject to evaluation, if a different organization. That person must also complete one of these forms, cross designating you.					
Associated Company / Organization Name #2 (write N/A if no partner)					
Title	First Name	MI	Last Name		Suffix
Street Address					
City		State	Zip	Country	
Phone	Fax		Email		

2

3

4

5

To be completed by suppliers of fingerprint template generator or matching software					
Responsible Party for supplier should complete these boxes					
Company / Organization Name					
Title	First Name	MI	Last Name		Suffix
Street Address					
City		State	Zip	Country	
Phone	Fax		Email		
Point of contact for day to day operations	Phone		Email		
Responsible Party for supplier of fingerprint software should identify the responsible party for supplier of smart card if any, and if a different organization. That person must also complete one of these forms, cross designating you.					
Associated Company / Organization Name #1 (write N/A if no partner)					
Title	First Name	MI	Last Name		Suffix
Street Address					
City		State	Zip	Country	
Phone	Fax		Email		

With my signature, I agree that this document is a sufficient description of the test to be conducted.

With my signature, I hereby request consideration as a Participant in the Minutia Interoperability Exchange Test II (MINEX II), and I am authorizing my company or organization to participate in MINEX II according to the rules and limitations listed in this document.

With my signature, I also state that I have the authority to accept the terms stated in this document

_____

SIGNATURE OF CARD SUPPLIER RESPONSIBLE PARTY DATE

_____

SIGNATURE OF FINGERPRINT SOFTWARE SUPPLIER RESPONSIBLE PARTY DATE